

IN THE CLAIMS:

1. (Original) A computer-implemented method for creating a graphical program that performs a numerical function, the method comprising:

displaying a node in a graphical program in response to user input;

configuring the node to receive data values, in response to user input;

configuring the node to perform a numerical function on at least a subset of the received data values, in response to user input;

executing the graphical program;

the node receiving a plurality of data values during execution of the graphical program;

the node determining a data collection on which to perform the numerical function, wherein the data collection comprises at least a subset of the data values received;

the node performing the numerical function on the data collection;

wherein the node maintains state information regarding received data values and uses the state information to determine the data collection on which to perform the numerical function.

2. (Currently Amended) The method of claim 1, further comprising:

receiving user input requesting to specify configuration information for the node;

displaying a graphical user interface (GUI) for specifying configuration information for the node, in response to the user input requesting to specify configuration information for the [[collector]]node;

wherein said configuring the node to perform the numerical function on at least a subset of the received data values is performed in response to user input received via the GUI.

3. (Original) The method of claim 1,

wherein the GUI is useable for specifying a collection mode for the node;

wherein the collection mode determines at least a subset of the received data values on which to perform the numerical function.

4. (Original) The method of claim 3,
wherein the GUI enables one or more of the following collection modes to be specified for the node:

Sliding Block;
Fixed Block; and
Always.

5. (Original) The method of claim 1,
wherein the node is a primitive node provided by a graphical programming development environment for inclusion in the graphical program.

6. (Original) The method of claim 1,
wherein the numerical function performed on the data collection comprises one of:

a numerical average function;
a summation function;
a minimum value function;
a maximum value function.

7. (Original) The method of claim 1,
wherein said configuring the node to receive data values comprises connecting an input terminal of the node to an output terminal of another node in the graphical program, in response to user input.

8. (Original) The method of claim 1,
wherein the node includes one or more output terminals corresponding to one or more numerical functions;

wherein said configuring the node to perform the numerical function on at least a subset of the received data values comprises connecting a first output terminal of the node to a data target in the graphical program, wherein the first output terminal corresponds to the numerical function.

9. (Original) A memory medium for creating a graphical program that performs a numerical function, the memory medium comprising program instructions executable to:

display a node in a graphical program in response to user input;
configure the node to receive data values, in response to user input;
configure the node to perform a numerical function on at least a subset of the received data values, in response to user input;

wherein during execution of the graphical program, the node is operable to:

receive a plurality of data values;
determine a data collection on which to perform the numerical function,
wherein the data collection comprises at least a subset of the data values received;
perform the numerical function on the data collection;
maintain state information regarding received data values and use the state information to determine the data collection on which to perform the numerical function.

10. (Currently Amended) The memory medium of claim 9, further comprising program instructions executable to:

receive user input requesting to specify configuration information for the node;
display a graphical user interface (GUI) for specifying configuration information for the node, in response to the user input requesting to specify configuration information for the [[collector]]node;

wherein said configuring the node to perform the numerical function on at least a subset of the received data values is performed in response to user input received via the GUI.

11. (Original) The memory medium of claim 9,

wherein the GUI is useable for specifying a collection mode for the node;

wherein the collection mode determines at least a subset of the received data values on which to perform the numerical function.

12. (Original) The memory medium of claim 11,
wherein the GUI enables one or more of the following collection modes to be specified for the node:

Sliding Block;

Fixed Block; and

Always.

13. (Original) The memory medium of claim 9,
wherein the node is a primitive node provided by a graphical programming development environment for inclusion in the graphical program.

14. (Original) The memory medium of claim 9,
wherein the numerical function performed on the data collection comprises one of:

a numerical average function;

a summation function;

a minimum value function;

a maximum value function.

15. (Original) The memory medium of claim 9,
wherein said configuring the node to receive data values comprises connecting an input terminal of the node to an output terminal of another node in the graphical program, in response to user input.

16. (Original) The memory medium of claim 9,
wherein the node includes one or more output terminals corresponding to one or more numerical functions;

wherein said configuring the node to perform the numerical function on at least a subset of the received data values comprises connecting a first output terminal of the node to a data target in the graphical program, wherein the first output terminal corresponds to the numerical function.